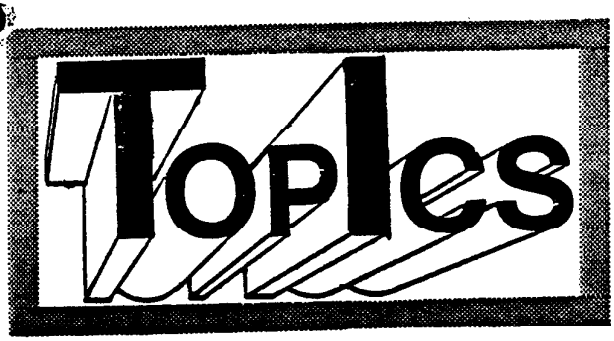
THE WORLD RETURNS TO LOS ANGELES

99'FEST—WEST'87
MAY 16—17, 1987

that's 16—17th
NOT 2—3rd!!!!



LA 99er COMPUTER GROUP Newsletter

VOL. 6 NO. 2 FEB. 1987

T 'n T

Terrie and I (Tom) plan to write this column together this year - you may never know who wrote what! First of all, I thank all of the members of this wonderful club for electing me to be President for this year. It is an honor which I am not sure I deserve, as my talents lie in other areas. Terrie as Vice-President will continue to help as only she knows how. And in that vein, I am taking the opportunity right now to thank Terrie for her marvelous leadership over the last three years. The LA 99'ers would not be in the good shape it is today, nor as well known or with such a good reputation, if it had not been for her prodigious efforts! The other new officers are also to be congratulated: Alan Whiteman will be Treasurer, taking over an impossible job after the excellent one done by Margaret Hutton over the past year. Our thanks to Margaret, also. And Doug Moore will be taking over as Secretary from Terrie Wilson, also deserving of thanks.

Most committee chairpersons will remain the same. Fred Moore has done an INCREDIBLE job with the Library, and I love him for it. Thank goodness he has agreed to continue! Chick De Marti and Alan Whiteman will stay on as his assistants. Gail Fair will again be chair of the Software Sales Committee and Joe Fierstein Chair of Equipment, as they have already done such a fine job. Ed May will be taking over from me as Membership Chairperson. As for the Newsletter, this will continue to be a group project - but Terrie will be "editor-in-chief."

And one last "appointment." George Steffen, who has contributed so much to this club, did not wish to have an official position, due to his illness. We are all pleased with his recovery to this point, and I hereby appoint George our "Resident Guru." He will continue to be active.

I have one statistic which may show the strength of our club. When I became Membership Chair last year, there were 137 names on the membership roll. 61 of those, for various reasons, have left the club, but in the same time we have gained 62 new members, from around the country and around the world. We have made international, as well as national, friends whom we never would have met otherwise. I hope many of you show up for the TI'FEST-WEST'87 this May!

(Terrie)

May 16 & 17 are the correct dates for FEST-WEST'87, I apologize for the incorrect dates printed prior to this. We are locked into the dates of the Show we tandem with Computer Sellathon and Expo. These are the dates the Shrine Exposition Hall has made available to the promoter. Again sorry for the confusion. Similar motel arrangements to last year are now being firmed. Any interested Vendor and/or Visitor is encouraged to communicate with me at, 148 S. Maple Drive, Beverly Hills, Ca. 90212.

It is with great sadness I read in the Muscle Shoals, Al. Newsletter of the loss of two of the finest and more prolific programmers to our community. JET John E. Taylor and Danny Michaels. Each for his own reasons has ventured to another computer. These two gents gave so very freely of their great talents, they will be greatly missed.

There is now surfacing a New generation of Programmers. Perhaps one or two of them are right there in your midst, look around you at the youngsters the 99/4A was purchased for. The one who started with modules and showed a curiosity beyond them. The 7-10 year olds of 3-5 years ago. Encourage them to follow their curiosity. These are our future Craig Millers, John Taylors, Danny Michaels, Don Thomsons. Don't let this talent slip away, it is there just needs nurturing. I see glimmerings of this in many of the Newsletters I read from across the world, Sacramento, Philadelphia, Tennessee. Some a bit overshadowed by grumps, and envy. Share the talents of your younger members through your Newsletters, the word will get out. We can learn from our young without a loss of face, try it you may like it!

We have recently had a good measure of success of matching sellers of systems with buyers, with thanks from both sides. For those who have no longer used programs and/or modules available, many times we have heard of people looking. Fest-West may have a consignment table if the need appears, in the meantime an ad in this Newsletter works very well.

The Raffle now held at each Club Meeting is being received with positive results, this money is designated

(cont. on Page 2@)

TWOS COMPLEMENT FBD6 AND ALL THAT

by Alan Whiteman

In last month's article on bits and bytes, Jim Swedlow referred to the use of Twos Complement in dealing with negative numbers. Jim noted that this can seem quite confusing. By definition, twos complement is obtained by adding a binary one to the ones complement - a piece of mathematical mystique that does little to help. However this can be better understood with a simple visual aid and examples to show why it is, in fact, useful.

Firstly, two complement is just one convention which we have agreed to use for representing negative integers. There are other conventions we could have chosen; the negative sign "-" is itself such a convention which is already familiar.

Secondly, we can think of negative numbers as having an opposite sense to their positive number counterparts - starting from zero the positive numbers represent one direction while the negataive numbers represent the other. In more familiar terms:

a) the elevation of Mt. McKinley is positive because it is above sea level, while that of Death Valley is negative because it is below.

b) the temperature in the living room might be a comfortable 20 deg. Celcius, while the freezer is a chilly 20 deg. below zero (-20).

In each case the zero is a reference or starting point from which the positive and negative directions take their significance. Similarly, in out car we could think of moving forward as being in the positive direction, while to move backwards would be negative.

If new acars really came with zero miles then the odometer would read exacly 00000. As we drive the odometer reads 00001, 00002 etc., until one day it reaches its maximum of 99999. We all know what happens next. If we continue the care has one hundred thousand mils on it but the odometer reads 00000 and starts again where we began.

Implicitly we have assumed that we were driving in the forward direction i.e. positive. What if we could drive backwafds with an odometer that runs in reverse and keeps track of our negative mileage? Starting again at zero, if we drive one mile backwards the odometer suddenly reads 99999 - a result we have seen before but with a quite different meaning. Continuing in reverse we would see 99998 after -2 miles, 99997 after -3, and so on.

But this is terrible! What would a reading of 60000 mean? It could be sixty thousand miles forward or forty thousand miles in reverse. Note that mathematical probless are not always suitable for road use! What if the odometer had already been around once or twice? We would be even less sure what it should mean.

Because we cannot have this uncertainty, we have a convention. Since we would like to be able to represent as many negative numbers as we do the positive numbers, let us agree to divide the total range available about in the middle. Let 00001 to 49999 represent the positive numbers, in that order, and 99999 to 50000 will be the numbers -1 to -50000. The positive and negative ranges are not quite equal since we need one space for our zero.

Now we are ready, at last, to get to the problem of twos complement! First we start a by thinking in hexadecimal words of two bytes and using our odometer as a visual aid, we get:

Hex	Decimal Value
>7FFF	32767
>7FFE	32766
.....	.....
>0002	2
>0001	1
>0000	0
>FFFF	-1
>FFFE	-2
.....	.....
8001	-32767
8000	-32768

We also agree under our convention that the maximum positive and negative numbers we will represent are >7FFF and >8000, 32767 and -32768 respectively. No more ambiguity.

Whiel providing negative numbers, twos complement is also efficient for our computers to process subtraction of numbers - it simply ADDS the two complement equivalent for the negative number! Consider the following examples:

```
>0014    20
>FFF6   -10
------  ---
>000A    10


>0014    20
>FFD6   -42
------  ---
>FFEA   -22
```

See how the addition of the twos complement on the left is equivalent to the subtraction of the decimal numbers on the right. Any overflow is ignored beyond the four hex digits.

Finally, to derive the twos complement of negative numbers, or to obtain the equivalent value for hexadecimal numbers above >8000, the process is the same. Add a binary one to the ones complement of the binary. Ones complement is equivalent to the logic function NOT and is obtained by replacing all binary ones with a binary zero and vice versa. For example:

-64 is >FFC0
(>0001 + NOT >0040 = >0001 + >FFBF = >FFC0

>FBD6 is -1066
(>0001 + NOT >FBD6 = >0001 + >0429 = >042A = 1066

Perhaps this is all still confusing, but at least Anglophiles will understand the title of this article.

ADDING TO YOUR SUPERCART

by Jim McCulloch (4/1/86)

As almost any 99'er knows, a Supercart is a solid state command module for the 99/4A which includes an Editor/Assembler GROM chip and a battery backable CMOS static RAM memory chip, the HM6264LP-15 by Hitachi or other makers. With this module, you can have power-up menu access to any Assembly Language program(s) you feel like including, as long as the total size doesn't exceed 8K.

The full instructions for the construction of the battery backed Supercart were described very well in MICROpendium Magazine in the June, July and August 1985 issues by its inventors, John Clulow and Ron Gries of the New Horizons Users Group of Ohio. It has come to my attention, however, that several people have earlier versions of the Supercart without the battery backup circuit (such as described in one of the Ryte Data newsletters). For anyone who doesn't have access to the MICROpendium issues (everybody should; it's a good magazine for us 99'ers in spite of the controversial major advertiser), this article will describe the addition of the battery backup circuit. For those of you who already have this most useful feature, read the next sentence. Thanks to Mark Lopez who told me how he did it, this article also describes the construction of a manually bank switchable module with TWO HM6264LP-15 chips installed for a total of 16K user addressable memory (albeit 8K at any one time). Although much of the following may be "old hat" to many of you, it includes some information I haven't seen distributed widely and so I naively include it so that more of us can share any helpful information.

PARTS LIST

PARTS _____ QTY
SOURCE _____ PRICE

(For the battery back-up circuitry)

1N914 Signal Diodes (2)
  Radio Shack 276-1122 $0.99 for 10
1K Resistor-1/4 Watt (2)
  Radio Shack 271-1321 $0.39 for 5
2.2 MFD Tantalum Capacitor (1)
  Radio Shack 272-1435A $0.59 each
Jumbo Red LEDs (1)
  Radio Shack 276-0141A $0.69 for 2
Lithium 3V Enercell Battery (1)
  Radio Shack 23-160 $1.79 each
Coin-Type Lithium Battery Holder (1)
  Digi-Key BH996-ND $1.25 each

(For the switchable second RAM chip capability)

HM6264LP-15 CMOS RAM IC (1)
  I.C.Express $2.95 each
1K Resistor-1/4 Watt (1)
  Radio Shack 271-1321 $0.39 for 5
SPDT Miniature Switch (Center Off) (1)
  Radio Shack 275-325 $2.19 each

(General Purpose)

Module Top 1953555-0107 (1)
  T.I.Parts Dept. $0.36 each
Module Bottom 1953554-0107 (1)
  T.I.Parts Dept. $0.30 each
Module Door 1015923-0007 (1)
  T.I.Parts Dept. $0.06 each
Door Spring 1015928-0001 (1)
  T.I.Parts Dept. $0.05 each
P.C.Board (2 sides) 1015921-0004 (1)
  T.I.Parts Dept. $3.60 each
E/A GROM Chip 1015960-1204 (1)
  T.I.Parts Dept. $3.60 each
(?C1?)Capacitor .1UF 50V 1501701-0122 (1)
  T.I.Parts Dept. $0.14 each
Screw 1010685-0034 (1)
  T.I.Parts Dept. $0.02 each

Radio Shack stores are generally ubiquitous. Digi-Key may be addressed at: P.O. Box 677, Thief River Falls, MN 56701-0988 or at 1-(800)-344-4539. I.C.Express can be addressed at: 15358 Valley Boulevard, City of Industry, CA 91746 or at 1-(800)-892-0809 ($10 minimum order). T.I. Parts Dept. can be reached at their new (1986) phone number at 1-806-741-3864 (Credit Cards useable and get faster response).

(See Figure 1 at end)

First of all, to open a module, unscrew the screw on the bottom cover. Then using a flat bladed screwdriver, pry the retaining lugs to be encountered in the bottom slots outwards. If you leave the foil label on the front of the module, it will act as the hinge of the clam shell thus formed, keeping proper alignment. You will probably find that the spring and door assembly are located on the top part of the module case. In some cases it's easier to reassemble the module if the spring and door are on the bottom. If you pinch the spring together (like a hose clamp), it will enlarge enough to fit over the larger post on the bottom module cover and the door fits in as before.

If you have an older style Supercart without the battery back-up circuitry, you may notice that pin 28 of the RAM chip is connected to the right hand side of the foiled hole F3. This is the +5V supply as it is supplied from the console via GROM port extender pin 19 (on the underside of the Printed Circuit Board). In keeping with the MICROpendium instructions, the foiled holes at both F3 and F1 should be electrically separated by scraping away the connecting foil. The C1 capacitor (the one which came installed on the board) should be moved to the right handed holes of F3 (which is +5V from the console) and F1 (which is Ground) if it isn't already there.

There should be a wire connecting RAM hole 28 to the left handed hole of F1 (as well as a connector from RAM

pin 20 to RAM pin 22 and a 1K resistor between RAM pin 20 and another part of the board.) The LED has a flattened side; the wire on this side should be connected to the left hand side of F3. This will supply + voltage to keep the CS2 pin of the RAM chip (pin 26) at a high voltage state which is needed for proper operation. To this wire in the left hand side of F3, connect one lead of a 1K resistor (R2 in Figure 2). The other end of R2 connects to any one of the Grounded foiled holes around the periphery of the board. The lead of the LED opposite the flattened side should be connected to any of the +5V holes available on the board; this may include the right hand side of F3 or any of the bottom right GROM holes as shown in Figure 2.

Somewhere on the board, you should make an electrically isolated hole; if you remove the 100 ohm reset resistor between F2 and R8, then the F2 hole is easily available by scraping away its foil connection with the rest of the board. Into this hole, solder a short length of stiff wire; to this terminal you will be soldering four other connections. The first of these four connections is the dark banded end of a diode (D1 in Figure 2); the other end of D1 is connected to a +5V supply hole, such as at the bottom right side of the right-most GROM socket. This will supply the +5V to activate the RAM chip and enable it to be written to. The second connection to F2 is one end of a 1K resistor (R3 in Figure 2). The other end of R3 should be soldered to the dark-banded end of another diode (D2 in Figure 2). The other end of D2 should be connected to the positive lead of the Lithium battery holder. (Soldering directly on Lithium batteries should be discouraged since there are reports of these types of batteries EXPLODING when subjected to heat or charging currents. The diodes in this circuit are apparently used to prevent such currents.) The third connection to F2 is the + lead (it's marked) of the Tantalum capacitor (C2 in Figure 2). The other lead of the Tantalum capacitor should be connected to any of the grounded holes around the periphery of the board. This is to "isolate" the power source. To any of the grounded holes available around the periphery of the board should be connected the negative lead from the Lithium battery holder. The fourth and final connection to F2 is a wire, the other side of which connects to Pin 28 of the RAM chip (disconnect any previous wiring from it to F3). Wrap any exposed wiring with electrical tape to prevent short circuits. I drilled a 1/4" hole in the front label side of the cartridge to let the LED shine through; it's not absolutely needed for correct functioning but it's a nice touch. Voila, this should do it for the battery backed circuit!

(See Figure 2 at end)

Adding a Second RAM Chip

This section describes how I added a second RAM chip by piggybacking it on top of the first. However, this makes the chip pile high enough so that the module cover will not close over it. Accordingly, I had to remove a small section of the top module cover (about 1 by 2 cm.) right at the point where it takes a couple of right angle turns. This is where the module narrows so that it will fit into the cartridge slot of the console. Since the chips take up some of this space, this "souped-up" Supercart needs to reside in a widgit or other cartridge expander (it even works well in a GK). To do the actual cutting of the module cover, I used an old soldering gun which had a plastic cutting tip but I suppose anything from drills to hot wires could be used also.

The Hitachi HM6264LP-15 is a 28 pin chip of which one pin is not connected, two pins are concerned with power supply (ground and +3-5V input),and 21 pins of which are address and data lines. This leaves 4 pins left over which control the functions of the chip. Pin 27 is the WE or Write Enable pin which determines whether the chip will be written to or read from and is controlled via the wire connected to edge connector 3; if the voltage to this pin is in a high state (+ voltage) then the chip's memory will be available to be read from whereas if it is low (0 voltage or grounded) then a write to memory is expected. Pin 26 is the CS2 pin which seems to act as a sensor as to whether power is applied or not; if this CS2 pin is at a low (0 voltage or grounded) state, then none of the chip's memory functions are accessable. This is why it is fed a continuous high voltage state via the LED which is connected to the +5V supply from the console (the left hand F3 hole connects with pin 26). Pin 22 is the OE pin or data bus in and I'm not entirely clear as to its meaning. However in this system, if this pin is at a high voltage state, output from the chip is disabled and if it is at a low state (0 voltage or grounded) then read and write functions can be done. The last of the four control pins is pin 20 or CS1 or chip select pin. When this pin is supplied with a high state (+ voltage) the entire chip pretends that it isn't there (it's "deselected"). When this pin is at a low state (0 voltage or grounded) then it gets the message that it has been "selected" by the rest of the system to converse with and its functions are enabled. If you look at the inside of a GK or Horizon Ramdisk which both use piggybacked 6264LP-15s, you will find pins 20 bent out with individual wires connecting them to the board; this is the way each chip is selected or deselected.

The above paragraph is probably boring and inaccurate but it helps to explain the circuitry necessary to add another RAM chip to the pile. It's relatively simple to piggyback another RAM chip on top of the first; bend in the pins to make a tight fit over the lower chip's pins by molding on a table top, then bend out pins 1, 2, 20, 27, and 28. Then solder the pins from the top chip to the bottom chip being careful not to make any solder bridges between adjacent pins. (In my module, I actually soldered the two together before I installed it on the board.) Pin 1 is ignored. Pins 2, 27, and 28 are connected to the same wires as supply the corresponding pins on the lower chip. If you connected all of the pins of both chips in parallel, you would have both chips doing the exact same thing - clones of each other. How do we give each chip its individuality? This is where the CS1 pins (pin 20) become useful. A "pullup" resistor is used to supply + voltage (a high state) to pin 20 of the chip not being used which as we read in the above paragraph has the effect of making that chip "invisible" to the system. In the absence of such a "pullup" resistor and + voltage source, these pins would tend to "float" down to a 0 voltage state which would

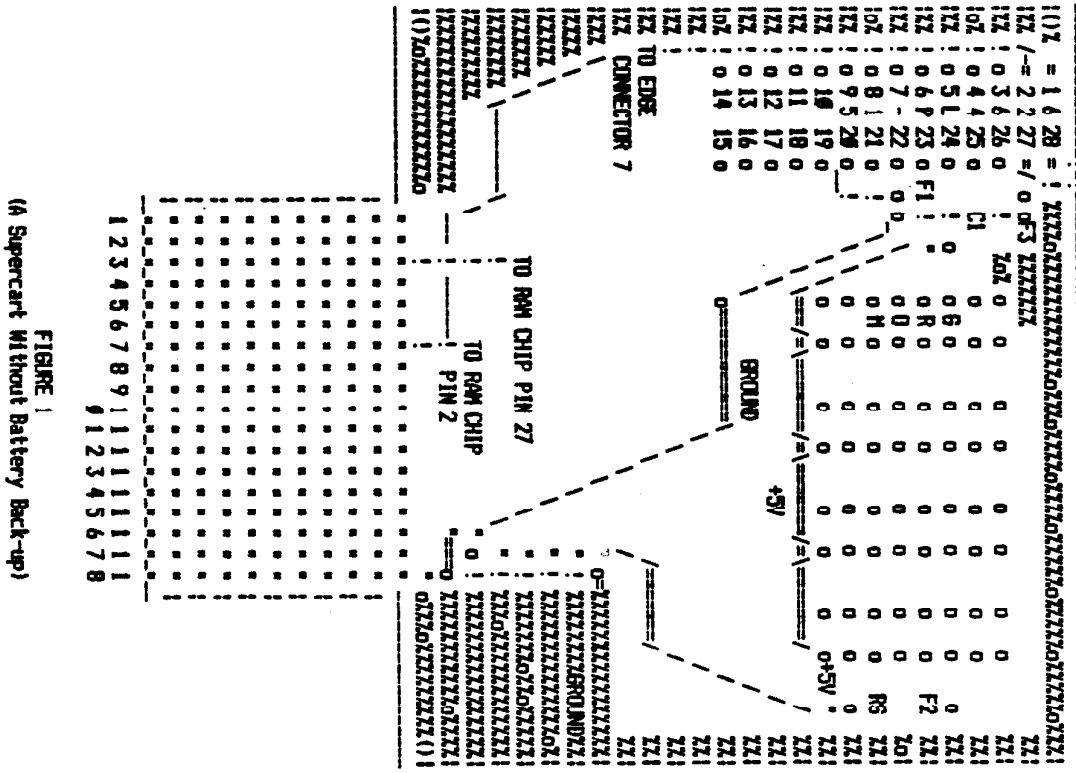
cause the system to "select" both chips at once. This would cause the system to read the same address of both chips simultaneously which would result in garbage and a probable crash. In the Supercart board, there is a resistor (R1) which acts as such a pullup resistor. In the version described for use in cartridge expanders, this R1 resistor is connected between CS1 (pin 29) and the +5V line from the console. This supplies a high state to deselect the chip. How then is the chip selected to enable it to do its thing? This is the function of the wire connecting pins 20 and 22 (the OE pin). When the OE pin is made a low state (0 voltage) then pin 20 is also made low since the resistor supplies voltage less readily than the direct connection to pin 22 "takes it away". To enable us to use both chips independently then, we could use a switch to connect the OE (pin 22) line to either of the RAM chips pin 20 while having pullup resistors connected to both pins 20 to keep the other chip deselected while the one chip is working.

This is exactly what I did: disconnect any wiring between pins 20 and 22 (to be found on the lower or older chip); next connect 1K resistors (R1 in Figure 3) between pin 20 and the +5V line for both the top and bottom RAM chips; next run wires from pins 20 of both the lower and upper chip to the outer terminals of the SPDT switch; then connect the center terminal of the SPDT switch to the OE pin with another wire (if you're tired of soldering on chip pins by now, you could run this wire to edge connector 2 which is the same line).

I then drilled another 1/4" hole in the front (label) side of the cartridge (somewhere on the left hand side to keep it away from the chips) to install the switch in. If the spring and door of the module cover have been moved to the bottom cover, it makes it easier to insert the modified board back into the module. Again, wrapping any exposed wires helps to prevent short circuits (in one of my earlier efforts, smoke rewarded me when I powered up the Supercart!) I finally used black electrical tape to wrap around the module and cover up the hole I'd made in the top cover. Voila, a manually switchable extra bank of usable memory! Now I can choose between 2 different entry menu screens simply by flipping the switch.

One other potentially useful feature I've found is this: with my previous single banked Supercart, I would more often than not scramble the memory if I removed the cartridge or inserted it with the console power on. (In retrospect, this is because the chip was hardwired to be constantly selected and was subject to transients and "spinal shock" when connected and disconnected.) Now if I "deselect" both RAMs by placing the switch in the center position, I can remove and insert the cartridge even with console power on without losing Supercart contents. To run, however, one or the other of the RAM chips has to be selected.

I hope these comments have been useful to any other "technoklutzes" beside myself out there. If anyone has any corrections or comments to make, I'd be pleased to get them at: Jim McCulloch, 9505 Drake Avenue, Evanston, IL 60203-1107 (CIS ID# 74766,580).


TO +5V SUPPLY
TO EDGE CONNECTOR 3
TO EDGE CONNECTOR 7
TO RAM CHIP PIN 27
TO RAM CHIP PIN 2
GROUND
+5V
C1
F1
F2
R8
1 2 3 4 5 6 7 8 9 1 1 1 1 1 1 1 1
0 1 2 3 4 5 6 7 8

Battery Back-up Circuit
On this page, Figure 1 shows the essential Supercart schematic.

FIGURE 1
(A Supercart Without Battery Back-up)

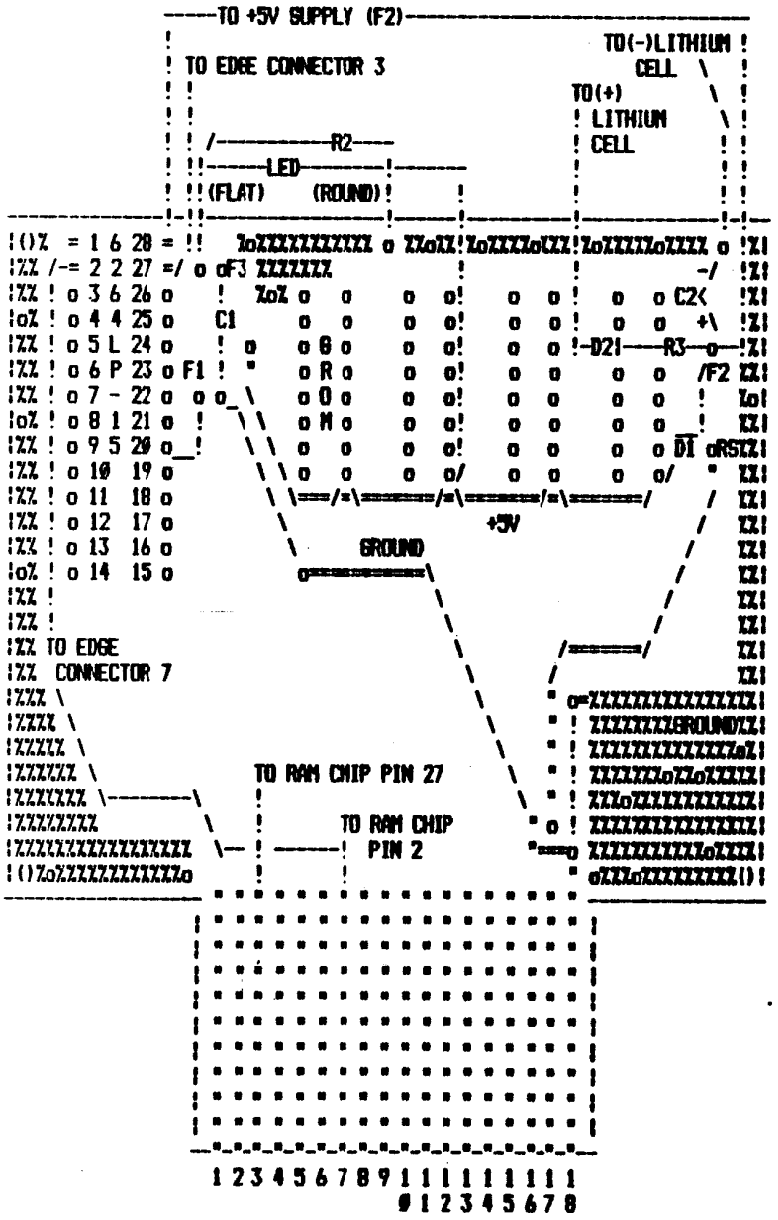


FIGURE 2
(Supercart With Battery Backup Circuit)

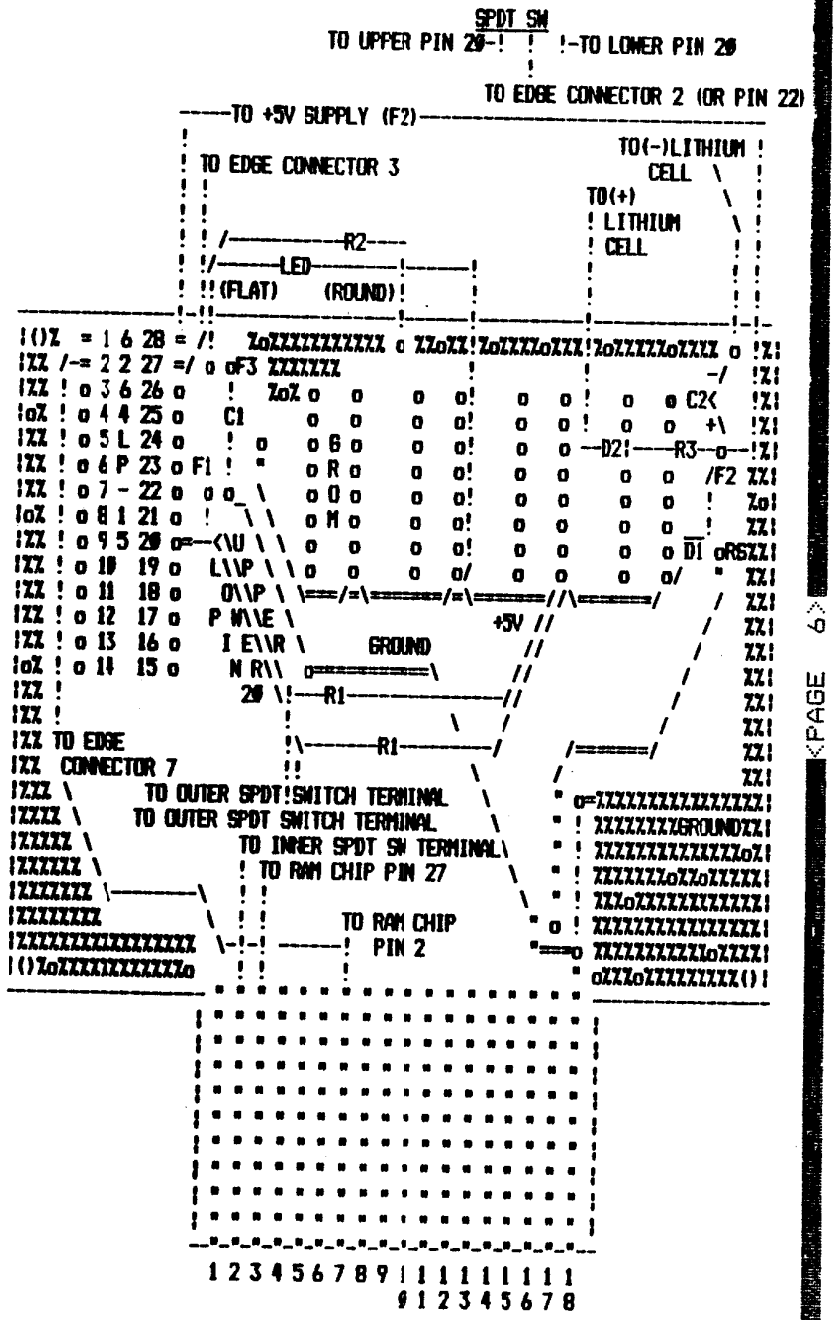


FIGURE 3
(Supercart With Switch Selectable RAM Chips)

Did you know that...?

by Chick De Marti

The Great ASCII Mystery

"All information that the computer uses is in the form of binary numbers. However, letters and other symbols are also used...(the computer in turn) is able to work with symbols because each one is given a numerical code...When you press the key for the letter "A" on your keyboard, 65, in a binary form is sent to the computer...Similarly, if the computer is to output an "A", it actually outputs a 65."

This brief explanation of how computers use the ASCII code system was found in the book, BEGINNING FORTH" by Paul Chirlian...but it should be of interest to those who use Basic and/or other languages.

Most of us are familiar with the codes from 32 (space) to 126 (curl), but what of some of the others? The ASCII numbers from 1 to 26 are control keys..."Control A has an ASCII code of 1, control B has an ASCII code of 2, etc." Some of these we come in contact with quite frequently without realizing it. ASCII 13, is called "carriage return"...is the ENTER or RETURN key. Others are CTRL G, ASCII 7 rings a bell, CTRL H, ASCII 8 represents the backspace. Your printer is controled by ASCII 10 (a line feed) and ASCII 12 (form feed) which advances the printer to the start of the next page. And as I said before ASCII 13 is the ENTER key. Concider this routine:

```
200 CALL KEY(0,K,S)
210 IF S=0 THEN 200
220 IF K<>13 THEN 220
```

The program will wait until the ENTER key is pressed.

"ASCII 27 represents escape. Sequences of ASCII codes that begin with 27 are called 'escape codes'." Printer owners are familiar with the sequence:

```
CHR$(27);"E" (for emphasized printing)
CHR$(27);"G" (to double strike the text)
```

Next month I'll give you some ideas on how to use ASCII codes to your advantage (although by now you must be getting the idea).

67 72 73 67 75 (CHICK)

IBM anyone?

(From the CALL SOUNDS newsletter...)

Q. I have an IBM PC at work and a TI-99/4A at home. Both are used mainly for word processing. Is there any way to transfer text from one to the other?

A. If you have a modem for each machine, and terminal emulatoar software such as PC TALK for the IBM and 4/A TALK or FAST-TERM for the 99/4A, you can use the capture buffer method and send from office to home and vice/versa. The text file can then be loaded into the woed processor for each machine. If there is no one at home (or at the office) when you are not there, you can join a service such as Compuserve. They give you a 128K of storage space free. When you are at home, load your text file into you CIS storage and unload it at the office when you get there. You can even carry your modem in your briefcase, if you have the proper cables, and use one modem for both computers.

<> <> <> <> <> <> <>

Want to use "@" and "&" ?

To change the TI Write "@" sign (shift 2), and the apersand from their double stike and underline functions, use a sector editor on FORMA1. Find the string 23 21 40 26 and change the 40 26 to 60 5C. Now FCTN C will double strike and FCTN Z will underline.

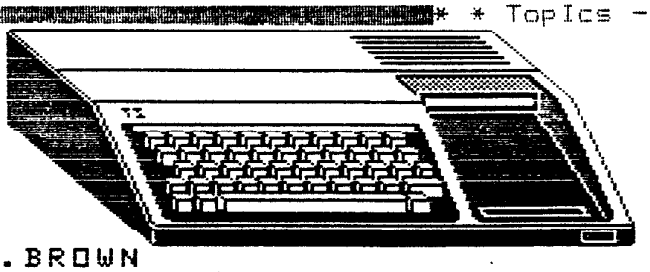
<> <> <> <> <> <> <>

(Can you figure out how I did the diamonds?)

(Answer...next page)

In BASIC (console or cartridge)

LIST 400 500 (without a dash in between) still works.



C. BROWN

A REMINDER (I just remembered)

(Actually, the Pittsburg U.G. newsletter, the "PERIPHERAL" reminded me!)
"Instead of using CALL CLEAR command to clear your screen, try these two variations:

```
10 CALL HCHAR(1,1,32,768)
```

This will clear your screen by sweeping from top to bottom.

```
10 CALL VCHAR(1,1,32,768)
```

This will clear your screen by sweeping from left to right."

(Plus...try

```
10 FOR I=24 TO 1 STEP -1
20 DISPLAY AT(I,1):"" :: NEXT I
```

This will clear your screen by sweeping from bottom to top...Chick)

<> <> <> <> <> <> <>

A safty valve in FORTH

II-WRITER TIP

From Mid-Hudson UG newsletter, Brett Kropf

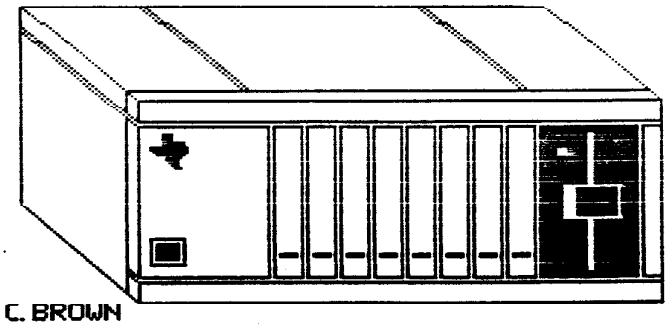
Don't like losing your filename?
When you want to insert a file into the text you are working on, you don't have to overwrite the original filename! When accessing the LoadF, PrintF, or SaveF editor command, instead of typing over the filename, just press Insert (FCTN-2) and type in the filename of the desired file and then at least one space. This will "push" the original filename to the right, while keeping it intact (unless you push it past the edge). When you go back to the SaveF, etc., simply Delete the inserted filename and you can then use the original filename(s) again without typing them in again, or better yet, trying to remember what it was!

<> <> <> <> <> <> <>

< HELP >

DOES ANYONE KNOW HOW TO GET OUT OF DM1000 - 3.3?

(Answer to the mysterious diamonds..."<0>"



C. BROWN

INFOCOM GAMES

From the San Antonio Area 99ers:
For the affecionados of the Infocam series of games, they are still releasing games for the TI99/4A. They know we are still here, so let them know you are still interested in their games.

<> <> <> <> <> <> <>

TEXAS INSTRUMENTS PHONE NUMBERS

TICARES	1-800-842-2737
TIRESPONSE..	1-800-232-3200
BUSINESS COM	1-800-847-2787
SOFTWARE ...	1-800-858-4075
TECHNICAL...	1-806-741-2663
PARTS	1-806-741-3064

TEXAS INSTRUMENTS is also selling either new or reconditioned parts at a resonable price. Seems like a bare bones P-Box is going for $90.00 and the controller for about $40.00. Call TI-CARES for more info and pricing.

Repair Facility address:
TEXAS INSTRUMENTS INC.
2305 NORTH UNIVERSITY AVE.
LUBBOCK, TEXAS 79415

ITEM.....	COST...	OHIO TAX	SHIPPING	TOTAL
CONSOLE	$30.50	1.68	3.00	$35.18
P.E. BOX	55.00	3.03	2.00	60.03
DISK DRIVES (Internal and external)	80.00	4.40	2.00	86.40
DSK CONTROLLER	44.00	2.42	2.00	48.42
32K CARD	44.00	2.42	2.00	48.42
RS 232	60.50	3.33	2.00	65.83
"P" CODE	33.00	1.82	2.00	36.82

<> <> <> <> <> <> <>

Well, I'm out of coffee...see you next month.

Chick

THE PHILADELPHIA AREA TI-99/4A USERS' GROUP (Dec. '86)

The program can be loaded from Basic, XB or Editor/Assembler, also from the CorComp Disk Manager.

The use of the E/A module is recommended, as it is sometimes required to load several modules of the program sucessively. Once loaded, the operator has the option of selecting screen color format and several other operating features by use of a DEFAULT program supplied by the author. (NOTE: Use the DEFAULT program with care, sometimes it interacts with the BBS program with weird results!) If you do not use the DEFAULT program, the same formats are operator selectable via keyboard commands.

Once into the program, you'll appreciate the ability to log all that takes place on disk, making review of messages "off-line" a reality. You can then compose your answers off-line and send them by using the convenient transmit buffer.

Another feature of the program is the automatic buffer dump to disk when full - meanwhile sending CONTROL S to stop the transmitting host during the buffer dump, and then sending CONTROL Q to signal the host to resume transmission. The operator has the ability to "freeze" the screen at any time and review what has happened to that point, and not miss anything while doing so. These and many more features are all in FAST-TERM, which is offered as FREEWARE.

Many DVUG members have this program. Make sure you send funds requested; that way authors are encouraged to continue to work within the TI environment.

THE FORUM

Thought for the month:
"The attention span of a computer is
only as long as its electrical cord"

Back when I was first learning about FORTH, Lee Stewart was writing words that impressed and inspired me to learn this "new" language. One of the few shortcomings was the lack of speech. Lee came up with a word to take care of that. This word takes a number off the stack which corresponds to the ROM location of the word to be said, and speaks that word. The list of the words and their locations can be found on pages 422-427 of the Editor/Assembler manual.

```
: SAY (n --- ) 4 0 DO 4 SRC DUP F000 AND 4 SRL 4000 = 9400 ! LOOP
               4000 9400 ! 5000 9400 ! DROP ; ( This word is in HEX )
```

I have this word on screen #3 of my system disk and I follow it with:

```
56B3 3A32 2D19 3793 6551 SAY SAY SAY SAY SAY
```

Have fun playing around with this word and don't forget to experiment! What happens if you specify a location where a defined word does not exist? Go ahead and try it and we'll see you next month.

THE PHILADELPHIA AREA TI-99/4A USERS' GROUP (Dec. '86)

TERMINAL EMULATORS
by Paul Wells

This article was excerpted from the Delaware Valley Users Group newsletter "The Data Bus".

Shortly after the First of the year 1985, a new terminal emulator program called FAST-TERM made its first appearance on the TI FORUM SIG of COMPUSERVE. Author Paul Charlton is a full-time student at Rensselaer Polytechnic Institute. Since Paul submitted this gem during the school year, one wonders how he made the time to accomplish this and maintain a passing average.

The program as originally published was a "bare bones" terminal emulator, but with a few "bells and whistles" that immediately produced wide-spread applause on the TI SIG. Things such as printer spooling and a wide selection of operating speeds, operator selectable color format of the screen, a 12K buffer and other goodies soon caused the word to be passed quickly to the members of the SIG. Since that time Paul Charlton has produced several other changes in the program which have made it even more valuable to the TI user.

Today, the FAST-TERM program has the ability to operate in either TE-II or XMODEM format; a feature, incidentally, not found on some of the other microcomputers on the scene today, who are still trying to implement XMODEM. You can use it now via FAST-TERM.

You've probably used the TE-II module and were aware of the slowness of the operation, but wait till you try the TE-II protocol built into FAST-TERM. It is approximately as fast as file transfer by X-BASIC, but it has the error checking of TE-II. (XB does not error check.)

The XMODEM protocol allows transfer of files in 128 character segements which is a 60% improvement over the 80 character limit imposed by TE-II. More and more downloads are appearing on COMPUSERVE in XMODEM format. A local BBS, Larry's Mousetrap is gearing up to implement XMODEM now, and the local FIDONET BBS's have had XMODEM for some time. What does this mean to you the TI user?

While you may not be able to download programs from the other local BBS's and use them without modification, you will be able to acess a lot of valuable information and data. (Oh incidentally, the IBM basic used on FIDONETS is about the simplest "foreign" computer language to interpet to TI Basic.)

Okay, what exactly does FAST-TERM do and require? It requires a disk drive and 32K memory. Additional drives and memory are nice, but not really required. A printer is also a nice extra, but you can do fine without it.

NINETY NINER NEWS

PURSUING PC-PURSUIT. . .

In this month's article, I was going to review products that we had on order. However, the only thing received was Millers Graphic's DISKASSEMBLER and I haven't had the time to take a good look at this new program yet. There are 2 reasons for this: 1. I spent the first two weeks of May in Gulfport Miss. with the Navy Reserves, and 2. I was selected to Market Test a new service called PC-PURSUIT.

PC-PURSUIT is a service of the GTE company. They own TELENET, a computer network that we use to access Compuserve. Several months ago, they added PC-PURSUIT. It was offered to people in 12 major cities in the US. They could use their computers and a local TELENET node and call *ANY* computer in any one of the other major cities. Fine for them, but not for us. Sioux Falls, SD isn't considered a major city (sigh.) However, they have expanded the area of callers by opening up any telenet node to be able to call the major cities. They even added two new cities. I suppose a list of the cities is in order now. Atlanta, Boston, Chicago, Dallas, Denver, Detroit, Houston, Los Angeles, New York, Philedelphia, San Francisco, Washington, DC. and the 2 new ones, Seattle and Newark.

The way to access the service is to call your local TELENET number, hit enter twice, enter d1. This puts you in the command mode of TELENET. Then type C DIAL617/12,(ID number) then enter. It will ask you for your password. Type it and press enter. If it isn't busy, it will say 617/12 connected. Then type ATZ to clear the 617/12 modem. Now you are ready to dial the number, say 664-5988. Type ATDT6645988 press enter and if the number is a computer and it answers you, CONNECT will be displayed on the screen. Now, use the host computer as you normally would.

A little explanation about some of the commands above. C-sets up TELENET to accept a command. 617-this is an example an area code that you want to call. /12-this is the baud rate (1200) that the other computer uses (it would be /3 for 300 baud.) ATZ-part of the Hayes compatible modem command set, means, ATtention, reset the modem back to it's original settings. ATDT-ATtention Dial Touchtone and then the 7 digit local number that you want to call.

This may sound a bit confusing and a lot of work, but I think that it is worth it. The charge for using this service is a flat $25 a month and there is a $25 sign-up fee. I have been getting long distance phone bills between $15 and $40 ever since I got my first modem. This is like heaven! It is a lot more relaxing than when you are paying the bill minute by minute, and there are quite a few TI boards that can be called. Following this article is an incomplete list of TI boards that I know are accessible through PC-PURSUIT. I will fill in the gaps as the info becomes available.

Thanks to Rory Binkerd for this interesting review.
(Continued on the next page.)

PC-PURSUIT....Continued:

Some of the nucances of the system are:

1. It is slower than a normal call to a BBS.

2. When the host computer does a check to see if you are running a TE2 cartridge (for special graphics, sound and music), it does not necognize the TE2. I haven't found the reason for this but suspect that the checking method will have to be changed to accomodate PURSUITISTS. The graphics, sound and music still work through PC-PURSUIT, it is only the check that fails.

3. With many users, it is common to find a certain area code busy. Each city has two types of lines, 1200 and 300. If one is busy, then try the other.

4. You are told the area code you can call, but not what prefixes are available. At this point, it's up to you to find out the prefixes. I hope they make a list available.

5. When calling the Washington, DC area code (202) you can reach computers in 3 different area codes (202, 703 and 301.)

For more information call 1 800 368 4215. With a modem, call 1 800 835 3001 300/1200.

Sorry if this seems a little sloppy, I did it in a hurry. I want to get back to them boards!!!!!!!!! -Rory Binkerd

<*>--<*>--<*>--<*>--<*>--<*>

And from BOB'S BASEMENT BULLETINS

ELEVEN NEW CITIES JOIN PC-PURSUIT...

I recently received a phone call from PC-PURSUIT. While I had them on the phone, I asked about the "new cities", that are coming to the network. The answer I received was that eleven new cities are scheduled to join the network, toward the end of the year (1986). They also told me the names of those cities which I have listed below.

1> Portland, OR
2> Tampa Bay, FL
3> Miami, FL
4> San Jose, CA
5> Glendale, CA
6> Minneapolis, MN
7> Salt Lake City, UT
8> Phoenix, AZ
9> Cleveland, OH
10> Milwaukee, WI
11> North Carolina (?? Chick)

NEW ADDS FOR FEBRUARY

FREEWARE

4110 MONTERULT $2.00 Freeware by Bob Guellnuty, X/B, 32K - Play Rouette as in Monte Carlo, excellent style, graphic and color, better know how to play or French, (SSSD)235

4114 TAX 86 $4.00 Freeware by James Sleeth, use multiplay module and disk to run. A excellent template printer for both Federal and State income tax forms. 2(SSSD)681

4115 RECORD PLUS Freeware by Steve Risner, X/B, 32K, A extremely versatile, memu driven data base disk. See flyer attached. (SSSD)360

EDUCATION EDUCATION

PRICES ARE MARKED AND INCLUDES DISK, A TWO AND ONE/HALF PERCENT (2 1/2%) DISCOUNT WILL BE GIVEN FOR EVERY ADDITIONAL EDUCATION DISK ORDERED. :: 1=0%, 2=2 1/2%, 3=5%, 4=7 1/2%, 5=10%, 6=12 1/1%, 7=15%, 8=17 1/2%, 9=20%, 10=22 1/2%, 11=25%, 12=27 1/2%, 13=30%, 14=32 1/2%, 15=35%, 16=37 1/2%, 17=40%, 18=42 1/2%, 19=45%, 20=47 1/2%, 21 AND OVER = 50% MAX.

Include $1.00 for every TWO disks mailed. I will be putting your ordered on SSSD disks (flippies) That will = 4 EDUCATIONS DISK PROGRAMS being mailed for $1.00. Number following ( ) are sectors used on disk.

2401 EDUCATION #1 $5.00 THE EVERYTHING TEACHER by Intellestar - Five sets of programs to provide customizing learning material for teachers and parents. TV SWEETAKES, BASEBALL, SPACE PATROL, LAST JELLYBEAN ON EARTH. Documentation included (paper) (SSSD)178

2402 EDUCATION #2 $5.00 THE TEACHER HELPER by Intellestar - A set of three programs. Highly responsive to teacher needs, featuring a fast flexible and easy to use Grade-Book and grade graphing system. Documentation included (paper)(SSSD)79

2403 EDUCATION #3 $5.00 TURN OFF THE WATER by Intellestar - A set of 2 program which provides customized spelling practice combined with entertaining "hangman" style game. You can create and store your own spelling list. good for grade level 2 thru 6. Documentation included (paper). (SSSD)48

2404 EDUCATION #4 $5.00 HEART ATTACK by Intellestar A Life Science program. age 11-adult. A 3 part program which puts you in control of a sophisticated model circulatory system, blood cells type and the nervous system. Documentation included (paper). (SSSD)59

2405 EDUCATION #5 $5.00 VYGER by Intellestar -Provides a relistic model of the Solar System. Learn about the atmospheres, surfaces, weather, temperature and navigational conditions pecular to each plantes ,the Sun and the moons. Documentation included (paper). (SSSD)97

2406 EDUCATION #6 $5.00 CELLS by Intellestar - A Life Science series.

- 3 parts set provides insight to the innermost working of life's building blocks the cells. Documenation included (paper). (SSSD)126

2407 EDUCATION #7 $5.00 INSIDE FRANKIE STIEN by Intellestar ,a Life Science Series. - A 2 part program try to keep Frankie alive by monitoring his body funtions and stimulate or suppress the various body systems. Documenation included (paper) (SSSD)56

2408 EDUCATION #8 $5.00 FIREBALL by INTELLESTAR - A arcade stlye action game which combines math practice with entertainment. Grade 2-8 Develop speed and accuracy (+.-,X,/).Documentation included (paper) (SSSD)54

2409 EDUCATION #9 $20.00 COMPLETE INTELLESTAR SET OF THE ABOVE 8 LIFE SCIENCE SERIES 2(SSSD)627 OR 1(DSSD)625

2410 EDUCATION #10 $2.00 EDUCATION A FREEWARE disk #4102 by Regena - 10 excellent programs music, typing, math, for the young (SSSD)316

2411 EDUCATION #11 $5.00 22 programs for the whole family by AMNION - CLOCK, COLOR CRAYON, VOCABULARY, TIME CLOCK, ADDITION TEST, LARGE CHARACTERS, WORLD FLAGS, MATH TEST(SPEECH), MATH TEST, FILE, GERMAN, MORSE CODE, SPANISH TRANSLATOR, TENSE, TRIANGLE, BIORHYTHMS, ALPHA, COLOR MATH, ALPHANUMBER, GRADING SYSTEM, SPELL (SSSD)355

2412 EDUCATION #12 $5.00 24 programs for the whole family by AMNION - WORD TOTAL, TRADE, BIORHYTHMS, PRECIOUS METALS CONVERSION, TIME AND SPEED, LEAST SQYARE, RESISTOR TEST, BIBLE BOOKS, CAREER FILE INDEX, PRESIDENTS II, CAPITOLS, STSTES, MATHATUTOR, FIRST ADDITION, SPELLING "B", CALL KEYS, DAYS BETWEEN DATES, DAYS WEEK YEAR, NAME NUMBER (SSSD)360

2413 EDUCATION #13 $5.00 18 educational programs for the whole family by AMNION -APELLING WORDS, WORD SWYLLABLE, WORLD FLAGS, QUADRATIC EQUATION, SPELL DOWN, SOLAR, DIVIDE, TEST TUBE, TRINOMIAL, AX BY = C, COLOR FRACTIONS, FLAGS OF THE NATIONS, FACTS OF LIFE, DRAW, ELEMENTARY ARITHMETIC, GUESS THE NUMBER, FINGLRS SPELLING,(SSSD)354

2414 EDUCATION #14 $5.00 14 programs for the whole family by AMNION - RELATIVE IQ TEST, ALGEBRA, BOWLING MATH, CHANGE, CATAPULT, PROJECTILE PROBLEMS, METRICE CLASS, GRADING SYSTEM, A TYPING TUTOR, JR HI MATH, TESTING, STATES QUIZ, COUNTRIES, SCIENCE, (SSSD)289

2415 EDUCATION #15 $5.00 12 excellent educational programs for the young at heart by AMNION - 5 programs of SPEAK SPELL FLASH, BASEBALL MATH, TRIGNOMIC SOLUTION OF TRIANGLES, WORD TEACHER, MATH RACE, MATH AID, GRADE BOOK, HAPPY SPELL, (SSSD)348

2416 EDUCATION #16 $5.00 14 Educational program for age 4 to 40 by AMNION - MICHIGAN, RUSSIAN TYPEWRITER, SPANISH MATH QUIZ, PIGGY BANK, EUROPEAN CAPITOLS, SPELLING TESTER, TEACHER'S GRADEBOOK, STATES & CAPITOLS, TEACHER'S CLAIM, TEST WRITER, LANGUAGE EXPRESSIONS, WORD SCRAMBLE, ADDITION PRACTICE, (SSSD)339

2417 EDUCATION #17 $5.00 16 educational program for the whole family by AMNION - WORD TWINS, ROADRUNNER, ALPHABET SOUP, SPELLING TESTER, LET'S PLAY ARITHMETIC, LEUKEMIA TUTORIAL, THE BARNYARD GROUP (5 programs add, count, divide, multi, & sub), JR HI MATH HELPER, CAPITALIZATION PRACTICE, USA FACTS, CONSTELLATIONS, (SSSD)348,

2418 **EDUCATION #18** $5.00 16 educational programs by AMNION - THE MUSIC TEACHER, GEOLOGY QUIZ, VOCABULARY TEST, STUDENT GEOMETRY AIDS (5 programs plane geometry, solid geometry I, solid geometry II, euclidian transformations, solution of triangles), SPELLING AIDS (2nd & 3rd), UNSCRAMBLE, FRACTION DRILL, ALPHASTER, IQ WORDMASTER, POETRY MAKER, (SSSD)353

2419 **EDUCATION #19** $5.00 15 educational programs by AMNION-NOUN INSTRUCTION, COUNTING, MATH QUIZZER, CAI TEST, SAY AND SPELL, ARITHMETIC DRILLS, GETTYSBURG ADDRESS, SUPER FLY, DRILL GAME, BIG LETTERS, MASTER WORD LIST, BIBLE GAME, MORSE CODE TRAINER, EVOLUTION, WATER POLLUTION STORY, NUMBER FUN (SSSD)354

2420 **EDUCATION #20** $5.00 15 educational programs by AMNION-STSTES QUIZ, PIGGYBANK II, THE ANSWER MAN, MATH HANGMAN, PRONOUN HIGHDIVE, SPANISH FLASH CARD, MATH CONCENTRATION, FRENCH VERBS, MAKE A STORY, ENERGY, NIGHT SKY. (SSSD)344)

2421 **EDUCATION #21** $5.00 13 educational program by AMNION-UP DOWN LEFT RIGHT, NUCLEAR CHEMISTRY, ROCKY ROBOT'S ALPHABET, ROCKY ROBOT'S DIVISION, SWEDISH VOCBULARY DRILL, MENUMATH, THE ABC SHOW, CHART OF ATOMOC WEIGHT, ALL ABOUT GEOEGIA, BEGINNING SPANISH, HEBREW KEYBOARD, THE PHYSICS RACE (SSSD)347

2422 **EDUCATION #22** $5.00 14 educational programs by AMNION-KATIE'S ADDITION, FRENCH GRAMMER GAMES, ROCKY ROBOT'S LEARN TO COUNT, ROCKY ROBOT'S SUBTRATION, PRESIDENT'S GAME, GEOMETRIC COORDINATES, SHAPE SEQUENCE IQ TEST, BIOLOGICAL SYMBOLS, LINEAR ALBEBRA, MENDEL'S LAW, 9th GRADE PHYSICS, CONSTELLATION, HANG GLOBE, TIMES FUNS GAME (SSSD)343

2423 **EDUCATION #23** $5.00 12 educatio nal programs from AMNION - TELLING TIME, PRE SCHOOL FUN, JEO AVEC LES, ADDITION MAGICIAN, MULTIPLICATION MADNESS, TYPIN G TRAINING, ROCKY ROBOT SINGS, READ FAS T, THE TYPING GAME, MATH MUNCHER (SSSD) 350

2424 **EDUCATION #24** $5.00 12 educational programs from AMNION - MORSE CODE T RAINING, CONCENTRATION SPANISH, SPACE S HUTTLE, ENGLISH TO CZECH, ARITHMETRIC D RILL, SPANISH VERBS, MAKE WORDS, VOLCAN O FACTS, TYPING, ADDITION MAGIC, ARITHG RAPH, MATH PUZZLE (SSSD)344

2425 **EDUCATION #25** $5.00 15 LANGUAGE educational programs -CZECH, FRENCH, GERMAN. HEBREW, RUSSIAN, SPANISH, SWEDISH , (SSSD)277

2426 **EDUCATION #26** $5.00 20 PRESCHOOL education programn - FIRST ADDITION, ABC, ALPHABET, NUMBERS, LETTERS, CLOCK, COUNTING, DRAWING, KEYBOARD, TELLINGT TIME, WORDS, (SSSD)346

2427 **EDUCATION #27** $5.00 8 EXCELLENT education program by John E.Taylor- great for pre-school children. COLOR MAST, EGG HUNT, INTRUDER, SPRIT BUILDER, ALPHABET DELIGHT, COAL TOCAS 2(SSSD)669

LIBRARIAN

FRED MOORE
7730 EMERSON AVE
LOS ANGELES
CALIFORNIA 90045

ERROR CHECK FOR XBASIC PROGRAM ENTRY

by Tom Freeman

Have you ever typed in a TI 99/4A version of a basic program from a magazine and noticed that the other versions have little numbers at the end of the lines that you don't have? They were for error checking on your typing, to insure no mistakes. Have you ever laboriously typed in a long program and run it, only to find that it crashes, or doesn't work as it is supposed to, all because of a simple typing error that you can't find? So why doesn't TI have one? NOW YOU DO!!

This may be the most useful program that I have published for general use, because almost everyone does basic programs at one time or another. It involves only one extra step for the programmer, and one for the user who is typing the published program in. It is really a rather simple method, and depends on the manner in which TI stores basic programs. Please note however that it requires a memory expansion and disk drive, and works only in Extended Basic (although Basic programs can be entered in XBasic, saved, and then run in Basic).

You may remember the format in which "MERGE" type programs are stored on disk. If you don't, see our article a couple of months back on the various formats in which programs are stored. The MERGE format is actually a duplicate of the way in which the actual program is stored in memory, or on disk, the difference being that it is a display type file, with each record starting with two bytes for the line number, and then the actual program line. In memory however, the program lines are stored contiguously, and in seemingly random order (actually the order depends on the order in which they were entered). A separate line number table is stored below the program area and keeps track of the line numbers and pointers to where each line begins. Now each line consists of one byte "tokens" for all reserved words (see the list I published last month) with all strings, including the names of subprograms such as LOAD, SCREEN, etc., being spelled out directly.

When you enter any line in XBasic (either a command, or a program line with the line number coming first) it is first moved to the so called "Edit Buffer" at address >8C8 in VDP. The basic bias is preserved. The purpose of this is that if you press FCTN 8 (REDO) then the whole line or lines can be retrieved. Next everything is "crunched" by replacing each reserved word with its token, subtracting the basic bias from strings, computing their length etc and placing the result in the "crunch buffer" at >820 in VDP. Once it is there, it can be transferred to the appropriate place in memory expansion. This is the area that is used when my program computes the "checksum" by merely adding the value of each byte! The number is never allowed to go over hex >FF - the high byte is ignored (thus, in decimal, no number over 255). The assumption is that is is extremely unlikely, probability approaching zero, that a small number of mistakes will result in a number that differs by exactly 256, or a multiple thereof. The one exception is if you transpose two characters - there's nothing I can do about that!

Now what does the programmer do? First his program must be completely debugged, as no changes can be made after the checksums are computed, or they will of course differ. Next he saves his program in merge format. Now the following program must be run on the result.

CHECKSUM

```
100 !CREATE CHECKSUMS FOR XB
ASIC PROGRAMS, BY TOM FREEMA
N, LA 99'ERS !250
110 !SHOULD BE USED TOGETHER
 WITH "CHECK" ASSEMBLY FILE
THAT WILL PRINT CHECKSUMS ON
 SCREEN !099
120 DISPLAY AT(2,1)ERASE ALL
:"CREATE CHECKSUMS FOR XBASI
C ERROR CHECKING": :"    by
Tom Freeman" !065
130 DISPLAY AT(10,1):"INPUT
MERGE FILE?":" DSK1." !067
140 DISPLAY AT(13,1):"OUTPUT
 MERGE FILE?":" DSK1." !198
150 ACCEPT AT(11,3)SIZE(-15)
BEEP:I$ :: OPEN #1:I$,VARIAB
LE 163,INPUT !192
160 ACCEPT AT(14,3)SIZE(-15)
BEEP:O$ :: OPEN #2:O$,VARIAB
LE 163,OUTPUT !053
170 DISPLAY AT(20,1):"ANALYZ
ING LINE":"CHECKSUM IS " !01
4
180 LINPUT #1:A$ :: IF LEN(A
$)=2 THEN CLOSE #1 :: PRINT
#2:CHR$(255)&CHR$(255):: CLO
SE #2 :: STOP !115
190 Z=ASC(A$)*256+ASC(SEG$(A
$,2,1)):: DISPLAY AT(20,15)B
EEP:Z !141
200 B$=SEG$(A$,3,163):: L=LE
N(B$):: IF L>157 THEN 230 !1
62
210 N=0 :: FOR X=1 TO L :: Y
=ASC(SEG$(B$,X,1)):: N=N+Y :
: NEXT X :: N=N AND 255 :: N
$=STR$(N):: N$=RPT$("0",3-LE
N(N$))&N$ !088
220 DISPLAY AT(21,13)BEEP:N$
 :: PRINT #2:SEG$(A$,1,L+1)&
CHR$(131)&N$&CHR$(0):: GOTO
180 !252
230 DISPLAY AT(22,1)BEEP:"WA
RNING!":" LINE";Z;"IS TOO LO
NG!":"PRESS ANY KEY TO CONTI
NUE" !123
240 CALL KEY(0,K,S):: IF S=0
 THEN 240 ELSE PRINT #2:A$ :
: GOTO 180 !232
```

Notice the "!" and 3 numbers at the end of each line? The program was run on itself! Here is what happens. Each record of the merge file is read in, the first two bytes ignored (we don't need the line number) and the rest are added up. Next the identical record is printed to the output file, with the addition of the token for ! (remark) and the 3 characters of the checksum. This will work even if the program line already contained a remark (as in lines 100-110). THE USER MUST BE WARNED NOT TO TYPE THESE 4 CHARACTERS, since they were not computed into the checksum. At the end (it may take a little while with a long program, but only needs to be run once) the programmer types NEW and merges in the ouput file, then saves it in normal mode, or lists it to printer, or whatever. This is the form to be published.

Now what the user must do is once type in the source code attached to the end of this article, and assemble it (a CALL LOAD version is also supplied for those who don't have the Editor/Assembler). If the object code created was called "CHECK" then he must type the following upon entry into XBasic: CALL INIT :: CALL LOAD("DSKx.CHECK") :: CALL LINK("CURSOR"). This one line with a line number can be saved on disk, and then RUN each time it is needed, rather than type the whole line. What the assembly routine at CURSOR does is some housekeeping such as moving the numbers 0-9 to character sets 13-14, changing the colors there, redefining the cursor, putting up the title screen etc, and then turning on the user defined interrupt. Now at every VDP interrupt (each 1/60 second) the routine at CHECK begins. The interrupt can

be turned off with CALL LINK("OFF") and back on with CALL LINK("ON") at any time, and the shape of the cursor will tell you which mode you're in. Now EVERY TIME you enter a new program line (and for some reason also after FCTN 8 - REDO even if no changes are made) the checksum will appear at the bottom of the screen and one extra line scrolled up. HERE IS THE KEY - IT SHOULD CORRESPOND TO THE ONE PUBLISHED THAT YOU ARE ATTEMPTING TO COPY IN. Hence, no errors!!!

I think the source code is sufficiently commented to explain what is going on. I must add that I spent many hours with Miller Graphics "EXPLORER", by Doug Warren, finding out WHAT is going on when you enter a line in XBasic. The address range in GROM of >6AA0 to >6AD0 should be broad enough to cover the various versions of XBasic out there, since they differ by a few bytes here and there (the actual range needed in my module was >6AAE to >6ACA. This area contains the loop where the first key press on entry of a new line is located. As soon as the first key is pressed then the GROM code moves on. I needed this area so as to reset the flag that indicates the checksum has been printed, in order to avoid having it printed again and again! Notice the fairly cumbersome method of peeking at the GROM address, which must then be reset, since just looking at it destroys it! I discovered that the line number entered is saved at <u>both</u> >8304 and >834A and only when it is at both is the crunch buffer finished being filled with the crunched line. If you are entering a direct command, >8304 is not used until much later, which is why I clear it at the beginning of each entry, so the routine won't get confused.

Finally if all the criteria are met (>8304=>834A and KEY(>8375) contains the valid entry key :enter >0D, up arrow >0B, or down arrow >0A) then the meat of the program goes to work, computes the checksum and puts it on the screen after an extra scroll (XBasic does its own scroll after I'm finished). Please note that I use BLWP @XMLLNK with data SCROLL instead of adding the whole routine. This saves a lot of typing. However for those of you who are interested I am also providing the entire routine done by DISKASSEMBER, so that you can place it in an E/A assembly file if you wish, as this one exists in Bank 1 of XBasic's ROM at >6000->7FFF, and hence can't be used by E/A.

I'm hoping that everyone finds this program useful and that it is widely used. I'm only sorry I didn't write it three years ago! Finally, I would like to thank Doug Warren for writing "EXPLORER" without which I could not have done this, since I needed to find out where XBasic does what! (I also must blame Doug for my bleary eyes!) And I especially would like to thank Craig Miller for his invaluable help and advice while I was writing the program. As Craig slowly leaves the TI community, we will all feel the loss.

CHECK OBJECT CODE - CALL LOAD VERSION

```
100 CALL INIT :: CALL LOAD(9
460,0,0,0,0,0,0,196,160,196,
216,0,10,11,13,0,0)!180
110 CALL LOAD(9404,0,126,66,
66,66,66,126,0,31,31,32,32,8
8,66,65,83,73,67,32,69)!144
120 CALL LOAD(9504,82,82,79,
82,32,67,72,69,67,75,69,82,3
2,32,32,32,32,32,85,83,73,78
)!107
130 CALL LOAD(9526,71,32,67,
72,69,67,75,83,85,77,83,32,3
2,32,32,32,66,89,32,84,79,77
)!119
140 CALL LOAD(9548,32,70,82,
69,69,77,65,78,44,32,76,65,3
2,57,57,69,82,83,2,132,0,10)
!052
150 CALL LOAD(9570,17,2,2,36
,0,7,2,36,0,48,192,68,2,33,0
,176,6,193,4,32,32,32)!199
160 CALL LOAD(9592,4,91,2,0,
3,240,2,1,37,4,2,2,0,8,4,32,
32,44,2,0,4,128)!121
170 CALL LOAD(9614,2,1,39,22
,2,2,0,80,4,32,32,44,2,0,7,0
,4,32,32,36,4,32)!166
180 CALL LOAD(9636,32,24,0,3
8,2,2,37,22,2,3,96,96,2,4,0,
36,192,66,172,131,6,4)!204
190 CALL LOAD(9658,22,253,2,
0,2,228,2,2,0,24,4,32,32,36,
4,32,32,24,0,38,2,0)!067
200 CALL LOAD(9680,2,228,2,1
,37,46,2,2,0,24,4,32,32,36,4
,32,32,24,0,38,2,0)!020
210 CALL LOAD(9702,2,228,2,1
,37,70,2,2,0,24,4,32,32,36,2
,0,3,240,2,1,37,12)!006
220 CALL LOAD(9724,2,2,0,8,4
,32,32,36,2,0,38,36,200,0,13
1,196,4,91,2,0,3,240)!119
230 CALL LOAD(9746,2,1,37,4,
2,2,0,8,4,32,32,36,4,224,131
,196,4,91,216,32,152,2)!239
240 CALL LOAD(9768,36,248,6,
224,36,248,216,32,152,2,36,2
48,6,224,36,248,6,32,36,248,
136,32)!133
250 CALL LOAD(9790,36,248,36
,250,26,8,136,32,36,248,36,2
52,27,4,4,224,36,244,4,224,1
31,4)!013
260 CALL LOAD(9812,216,32,36
,248,156,2,6,224,36,248,216,
32,36,248,156,2,2,0,8,28,2,1
)!054
270 CALL LOAD(9834,37,20,2,2
,0,2,4,32,32,36,2,0,8,15,2,1
,244,0,2,2,0,13)!105
280 CALL LOAD(9856,4,32,32,3
2,5,128,6,2,22,251,2,0,7,4,4
,32,32,48,7,96,36,244)!204
290 CALL LOAD(9878,22,62,2,1
,0,3,152,33,36,254,131,117,1
9,3,6,1,22,250,4,91,200,32)!
180
300 CALL LOAD(9900,131,4,131
,4,19,49,136,32,131,4,131,74
,22,45,7,32,36,244,200,160,1
31,66)!038
310 CALL LOAD(9922,9,130,2,0
,8,32,2,1,39,22,4,32,32,44,4
,224,37,2,184,49,37,3)!195
320 CALL LOAD(9944,6,2,22,25
2,200,11,36,246,4,32,32,24,0
,38,2,0,2,226,193,96,37,2)!1
38
330 CALL LOAD(9966,2,2,0,10,
2,3,0,100,2,6,0,2,4,196,61,3
,6,160,37,94,5,128)!027
340 CALL LOAD(9988,192,194,6
,6,22,248,193,5,6,160,37,94,
194,224,36,246,4,91)!104
350 CALL LOAD(16376,79,78,32
,32,32,32,37,244)!042
360 CALL LOAD(16368,79,70,70
,32,32,32,38,14)!240
370 CALL LOAD(16360,67,72,69
,67,75,32,38,36)!002
380 CALL LOAD(16352,67,85,82
,83,79,82,37,122)!053
390 CALL LOAD(8194,39,22,63,
224):: CALL LINK("CURSOR")!1
43
```

SCROLL SUBROUTINE - FOR USE IN OTHER PROGRAMS

```
* WORKSPACE MUST BE >83E0
SCROLL LI   R12,>02E0
       LI   R10,>0020
       CLR  R9
       MOV  R11,R6
       BL   @AA
       LI   R5,>8C00
       LI   R4,>02E0
       LI   R1,>7F80
       LI   R2,>001C
       BL   @AF
       MOVB R1,*R5
       SWPB R1
AB     MOVB R1,*R5
       DEC  R2
       JNE  AB
       SWPB R1
       MOVB R1,*R5
       MOVB R1,*R5
       B    *R6
AA     CLR  R8
       MOVB @>83F5,*R15
       STWP R7
       MOVB R10,*R15
AD     MOVB @>8800,*R7+
       INC  R10
       INC  R8
       DEC  R12
       JEQ  AC
       CI   R8,>000C
       JLT  AD
AC     MOVB @>83F3,*R15
       ORI  R9,>4000
       MOVB R9,*R15
       STWP R7
AE     MOVB *R7+,@>8C00
       INC  R9
       DEC  R8
       JNE  AE
       MOV  R12,R12
       JNE  AA
       B    *R11
AF     MOVB @>83E9,*R15
       ORI  R4,>4000
       MOVB R4,*R15
       NOP
       MOVB R1,@>8C00
       B    *R11
```

```
* SOURCE CODE TO WRITE CHECKSUM FOR ENTERED XB LINE ON SCREEN
* BY TOM FREEMAN, LA 99ERS
* THIS IS PUBLIC DOMAIN, PLEASE DISTRIBUTE IT WIDELY!
       DEF  ON,OFF,CHECK,CURSOR
VMBR   EQU  >202C
VMBW   EQU  >2024
VSBR   EQU  >2028
VSBW   EQU  >2020
VWTR   EQU  >2030
XMLLNK EQU  >2018
SCROLL EQU  >0026          ADDRESS OF ROUTINE IN ROM INDEXED ON >6010
NSAVE  EQU  >8304                EQU >7ADA IN MY XB MODULE
LSAVE  EQU  >8342          ADDRESS WHERE LENGTH OF CRUNCHED LINE IS SAVED
FAC    EQU  >834A
GRMRA  EQU  >9802          GROM READ  ADDRESS PORT
GRMWA  EQU  >9C02          GROM WRITE ADDRESS PORT
DONE   DATA 0
SAV11  DATA 0
SAVEGA DATA 0
LOWAD  DATA >6AA0          /ADDRESS RANGE IN GROM WHERE FIRST KEY PRESS
HIAD   DATA >6AD0          \ON COMMAND LINE IS REQUESTED
ENTER  DATA >000A,>0B0D    ENTER KEY, UP AND DOWN ARROW
COUNT  DATA 0
CUR1   BSS  8
CUR2   DATA >007E,>4242,>4242,>7E00 HOLLOW CURSOR DATA
INVVID DATA >1F1F          INVERSE VIDEO COLORS,THIS IS BLACK ON WHITE
TITLE1 TEXT '  XBASIC ERROR CHECKER  '
TITLE2 TEXT '    USING CHECKSUMS    '
TITLE3 TEXT 'BY TOM FREEMAN, LA 99ERS'
GETDEC CI   R4,10          /IF NUMBER IS 10+ THEN NEED TO GET TO >41 ("A"
       JLT  GD             \NOT >3A
       AI   R4,7
GD     AI   R4,>30         MAKE IT AN ASCII CHARACTER
       MOV  R4,R1
       AI   R1,>B0         THIS IS BASIC BIAS OF >60 PLUS >50 TO GET TO
       SWPB R1             TO MSB    ALTERNATE CHARACTER SET AT ASCII 128
       BLWP @VSBW          WRITE ON SCREEN
       RT
CURSOR LI   R0,>03F0
       LI   R1,CUR1
       LI   R2,8
       BLWP @VMBR          SAVE ORIGINAL CURSOR PATTERN AT CUR1
       LI   R0,>4800       /THE 80 BYTES FROM >4800 TO >4CF ARE ASCII 48-
       LI   R1,LBUF        !57 ("0" TO "9").  TEMPORARILY STORED AT
       LI   R2,80          \LBUF
       BLWP @VMBR
       LI   R0,>7000
       BLWP @VMBW          NOW PUT THEM AT >7000 AS ALTERNATE CHAR. SET
       BLWP @XMLLNK
       DATA SCROLL         SCROLL UP 1 LINE
       LI   R2,TITLE1
       LI   R3,>6060       ADD BASIC BIAS TO TITLE CHARACTERS
       LI   R4,36
       MOV  R2,R1
CR1    A    R3,*R2+
       DEC  R4
       JNE  CR1
       LI   R0,>2E4
       LI   R2,24
       BLWP @VMBW          WRITE 1ST LINE
       BLWP @XMLLNK
       DATA SCROLL         SCROLL AGAIN
       LI   R0,>2E4
       LI   R1,TITLE2
       LI   R2,24
       BLWP @VMBW          WRITE 2ND LINE
       BLWP @XMLLNK
       DATA SCROLL         SCROLL AGAIN
       LI   R0,>2E4
       LI   R1,TITLE3
       LI   R2,24
```

```
        BLWP @VMBW          WRITE 3RD LINE
* CALL LINK("CURSOR") DOES THE SETUP AND CONTINUES ON TO "ON"
* CALL LINK("ON") STARTS HERE AND DOESN'T NEED THE SETUP
ON      LI   R0,>03F0
        LI   R1,CUR2
        LI   R2,8
        BLWP @VMBW          LOAD THE HOLLOW CURSOR INTO VDP
        LI   R0,CHECK       LOAD THE INTERRUPT ADDRESS INTO THE ISR
        MOV  R0,@>83C4      \(INTERRUPT SERVICE ROUTINE) HOOK AT >83C4
        RT
OFF     LI   R0,>03F0
        LI   R1,CUR1
        LI   R2,8
        BLWP @VMBW          RELOAD THE ORIGINAL CURSOR
        CLR  @>83C4         CLEAR THE ISR HOOK (TURN OFF INTERRUPT)
        RT
CHECK   MOVB @GRMRA,@SAVEGA "PEEK" AT THE CURRENT GROM ADDRESS AND SAVE
        SWPB @SAVEGA          IT AT SAVEGA, MSB 1ST. GROM ADDRESS IS NOW
        MOVB @GRMRA,@SAVEGA INDETERMINATE
        SWPB @SAVEGA
        DEC  @SAVEGA          ADJUST FOR AUTO INCREMENT
        C    @SAVEGA,@LOWAD TEST FOR THE LOW END OF RANGE WHERE START OF
        JL   CHECK1           COMMAND LINE IS,JUMP OUT IF TOO LOW
        C    @SAVEGA,@HIAD  HIGH END OF RANGE
        JH   CHECK1           JUMP OUT IF TOO HIGH
        CLR  @DONE          RESET FLAG FROM PREVIOUS CHECKSUM ROUTINE
        CLR  @NSAVE         THIS CORRECTS FOR A MYSTERIOUS ERROR I FOUND!
CHECK1  MOVB @SAVEGA,@GRMWA RESET GROM ADDRESS THROUGH GRMWA PORT
        SWPB @SAVEGA
        MOVB @SAVEGA,@GRMWA
*NEXT 4 LINES SET THE "INVERSE VIDEO" FOR CHECKSUMS-CAN BE DELETED
        LI   R0,>81C        RESET COLORS FOR CHARACTER SETS 13-14 AT EVERY
        LI   R1,INVVID      INTERRUPT(XB ALWAYS RESETS TO DEFAULT). DELETE
        LI   R2,2           THESE 4 LINES IF YOU DON'T LIKE THE INVERSE
        BLWP @VMBW          VIDEO EFFECT
*NEXT 10 LINES CHANGE SCREEN & CHAR COLORS WHILE IN CHECKSUM MODE
*AND CAN BE DELETED IF YOU DON'T LIKE THE EFFECT
        LI   R0,>80F        START OF COLOR TABLE FOR CHAR SET 0
        LI   R1,>F400       WHITE ON BLUE
        LI   R2,13          13 COLOR SETS
COL     BLWP @VSBW          WRITE A BYTE TO COLOR TABLE
        INC  R0             NEXT COLOR SET
        DEC  R2
        JNE  COL
        LI   R0,>0704       SCREEN COLOR 4(DARK BLUE)
        BLWP @VWTR
*END OF OPTIONAL LINES
        ABS  @DONE          /IF THE ROUTINE WAS ALREADY DONE
        JNE  RETURN         \GET OUTTA HERE!
        LI   R1,3            CHECK FOR THE 3 VALID ENTRY KEYS AND LEAVE IF
CHECK2  CB   @ENTER(R1),@>8375  THERE AREN'T ANY. NOTE USE OF INDEXING
        JEQ  C1             IF VALID KEY THEN GO ON
        DEC  R1             GO FOR MORE
        JNE  CHECK2
        RT
C1      MOV  @NSAVE,@NSAVE /WHEN >8304 CONTAINS A NON ZERO KEY AND IS =
        JEQ  RETURN          \WHAT IS IN >834A THEN WE'RE READY TO GO!
        C    @NSAVE,@FAC
        JNE  RETURN
        SETO @DONE          INDICATE THE CHECKSUM IS ABOUT TO BE WRITTEN
        MOVB @LSAVE,R2      GET THE LENGTH BYTE OF CRUNCHED LINE
        SRL  R2,8           MOVE TO LSB
        LI   R0,>0820       CRUNCH BUFFER
        LI   R1,LBUF        WHERE WE WILL STORE IT
        BLWP @VMBR          MOVE IT
        CLR  @COUNT         COUNT WILL CONTAIN CHECKSUM, IN BINARY
C2      AB   *R1+,@COUNT+1 /ADD EACH BYTE OF CRUNCHED LINE TO IT, 1 BY 1
        DEC  R2             !BECAUSE WE ARE ADDING BYTES,WHEN WE GO OVER
        JNE  C2             \FF, THE CLOCK GOES BACK TO ZERO
DO      MOV  R11,@SAV11     SAVE THE RETURN ADDRESS
        BLWP @XMLLNK
```

```
        DATA SCROLL         SCROLL UP THE SCREEN
        LI   R0,>2E2        3RD COLUMN, BOTTOM ROW OF SCREEN
        MOV  @COUNT,R5      MOVE THE VALUE AT COUNT (WORD VALUE BUT LESS
        LI   R2,10          THAN 256, TO R5
        LI   R3,100         R2 AND R3 CONTAIN THE DIVISORS
        LI   R6,2           2 LOOPS FOR 100'S AND 10'S PLACE
D1      CLR  R4             ASL DIVISION IS DONE THIS WAY.VALUE OF 1ST R
        DIV  R3,R4          IS DIVIDED "INTO" 2ND R(E.G. R3 INTO R4). THE
*                           2ND REG IS ACTUALLY 2 CONTIGUOUS REGISTERS.
*                           THE QUOTIENT IS PLACED IN THE FIRST AND THE
*                           REMAINDER IN THE 2ND.ORIGINALLY THE FIRST MUST
*                           BE 0, OR THERE WILL BE AN "OVERFLOW"
*                           SO R4 NOW CONTAINS THE INTERGER QUOTIENT
        BL   @GETDEC        CONVERT IT TO ASCII AND PUT ON SCREEN
        INC  R0             NEXT SCREEN POSITION
        MOV  R2,R3          NEXT DIVISOR
        DEC  R6             ANY MORE TO DO?
        JNE  D1
        MOV  R5,R4          1'S PLACE IS THE REMAINDER FROM 2ND DIVISION
        BL   @GETDEC        PUT THIS ONE ON SCREEN TOO
        MOV  @SAV11,R11     RESTORE RETURN ADDRESS
RETURN  RT                  AND RETURN
*                           THIS IS END OF PROGRAM AND IS A CONVIENT PLACE
*                           TO PUT THE BUFFER,WHICH HAS NO DATA TO START
LBUF    END
```

(cont. from Page 1)
to purchase of hardware for our BBS, support it.

As you can see this is no longer Terries Corner. We have new Officers and Tom Freeman is our new President. Tom and I participate in the Sunday night Conference on GEnie (6P PST) we have been nicknaming it as TnT so we are renaming this page as TnT (dynamite duo). Well that sort of continues the way it has been. Tom has become a very trusted friend and has had for quite some time now influence in TopIcs, this legitimizes it. We do not always agree and that is OK, the success of this Club is its excellent members along with a group of Officers who respect and like one another, differences of opinion, yes for sure. Back stabbing and/or infighting not at all. We function with pride, all of us. Tom has some very interesting statistics on our membership, hopefully he will include it in his commentary.

Food drive of Christmas was VERY successful, combined with a very generous money donation by an East Coast member. The food was given to APLA (Aids Project L.A.), the money to PARA LOS NINOS (a haven for Children of Skid Row Parents). We have letters of appreciation from both groups, and space permitting we will print them. My thanks to the generosity of our Members.

Within this Newsletter is an ad from a very good friend of the TI community, Helene LaBonville of HAC LABS INC. Helene is offerring some excellent prices on Hardware and Software. Look the flier over very carefully, there is sure to be something there you need. We strongly urge support of the HONEST dealers among us.

Our Booth at the February Sellathon as usual introduced us to new friends and gave us the opportunity to find new products for our Marketplace. This time we are very happy to say we will now have Prometheus 1200B 300/1200 auto ans/auto dial, Hayes Compatible external modems for $145.00. Normal price $299. Not bad, our thanks to Dean Conrad for bringing this to our attention. Our Chicago friends should contact Dean for this, he resides there now. WE will also have excellent prices on paper goods, computer post cards and index cards among others. Look for this next month.

Guess that is it for now, I thank all who have worked together this last year, and welcome those joining us this year. If you look closely you will see we are all still there in one capacity or another. New blood, new directions with the support of ALL of us. If you have not attended a meeting lately, come and see how this has improved our meetings. New format, new ideas, all appreciated and welcomed. In this issue for our members is included the 1987 Library List. A tremendous job done by the Number 1 Librarian, Fred Moore. Fantastic!!! Thanks Fred.

CLUB OFFICERS

President:
Tom Freeman
454-1943

Vice-President:
Terrie Masters
271-6930

Secretary:
Doug Moore
451-1069

Treasurer:
Alan Whiteman
379-8031

Librarian:
Fred Moore
670-4293

Library Assistants:
Alan Whiteman
Chick De Marti
532-8499

Software Sales:
Gail Fair
326-6660

Equipment:
Joe Fierstein
377-9834

Membership:
Ed May
644-6241

Programs:
Fletcher Wicker
325-9151
Steve Chalcraft
494-3953

TopIcs Editor (in-Chief)
Terrie Masters

Resident Guru
George Steffen
329-3527

MARKETPLACE

(the marketplace is a fund raiser for the club, that is, the "profit" goes to maintain the quality of this News-letter. In general the price listed splits the difference between cost and retail. Please help your Club.
PLEASE ADD $1.00 S&H CA. RES 6.5%TAX)

MILLERS GRAPHICS	
DISKASSEMBLER	18.50
ORPHAN CHRONICLES (priceless)	9.95
ADVANCED DIAGNOSTICS	18.50
NIGHT MISSION	18.50
GK UTILITY I	10.00
SMART PROGRAMMING FOR SPRITES	6.25
NEW RELEASES	
PRE-SCAN IT! (J.PETER HODDIE)	10.00
GRAM PACKER "	10.00
FONT WRITER "	19.00
PRINTER'S APPRENTICE (M.McCANN)	19.00
MYARC	
RS232	82.00
D/D DISK CONTROLLER	155.00
128K RAM DISK/SPOOLER	175.00
512K RAM DISK/SPOOLER	280.00
EXTENDED BASIC II LEVEL IV	80.00
128K RAM DISK W/XBASIC II	235.00
512K RAM DISK W/XBASIC II	340.00
INSCEBOT	
TI-ARTIST	17.00
DISPLAY MASTER	12.00
ARTIST EXTRAS	6.00
MEGATRONICS	
EXTENDED BASIC II PLUS	72.50
INTERN (BOOK ON GPL)	16.50
128K GRAM CARD	227.50
HARDWARE & SUPPLIES	
TEAC 55BV DSDD DRIVES	95.00
DISKETTES DSDD	.50
64K EPSON INT. PRINT BUFFER	45.00
COLOR RIBBONS (EPSON)	4.00
PROMETHEUS 300/1200 MODEM	145.00
BACK ISSUES	
SUPER 99 MONTHLY	1.25
MICROPENDIUM	1.25
SMART PROGRAMMER JUNE 1986	1.50
BEST OF NEWSLETTERS W/DISK	5.00
FORTH NOTES VOL 1-5 (2.50 EA)	10.00
BEGINNER'S FORTH NOTEBOOK	2.50
ASSEMBLY NOTES VOL 1	2.50
TECHNICAL AND BUSINESS BOOKS	5.00
SAMS BOOKS (VARIOUS)	5.00
SAMS BOOKS WITH CASSETTES	7.50

(please send your order to the CLUB address, not the Librarian, and add $1.00 per disk for postage and handling. CA residents add 6.5% sales tax)

AIDS PROJECT LOS ANGELES

3670 Wilshire Boulevard, Suite 300 Los Angeles, California 90010 (213) 738-8200

Board of Directors

Hon. Stephen M. Lacha
Chair

Matt Redman
Vice-Chair

Rev. A. Stephen Pieters
Secretary

Deborah Rutter
Assistant Secretary

Dennis L. Duban
Chief Financial Officer

Roberta Bennet
Mark Canton
Eunice Diaz, M.S., M.P.H.
Laurel E. Dickranian
Shelby Dietrich, M.D.
Max Drew
Richard Dreyfuss
Stuart Fink
Stephen Gabin, M.D.
Rt. Rev. Oliver B. Garver, Jr.
Jackie Goldberg
David E. Gooding
Jask Hamilton, M.A., M.F.C.C.
Barry Kroet
Gene La Pietra
Alexandra Levine, M.D.
Leonard Peterson
Peter Scott
Larry Sprenger
Cynthia Telles-Jimenez, Ph.D.
Joel Weisman, M.D.
Acenio Zapata, Ed.D., Dr. P.H.

January 5, 1987

L.A. 99er Computer Group
P.O. Box 3547
Gardena, CA 90247-7247
Att: Teresa Masters, President

Dear Friends,

Thank you for donating groceries to the Food Distribution Program of AIDS Project Los Angeles. Your contributions have been distributed to people with AIDS and ARC who depend on our program for supplemental food. We are currently serving over 400 clients a week.

As you probably know, the food program is supported entirely through community donations. Without the help of people like you, we would never be able to accomplish this monumental task. Your support not only nourishes people with AIDS and ARC, but enriches their lives as well.

On behalf of the people who rely on the Food Distribution Program, I thank you again for your support and good wishes.

Sincerely,

Jamie Henderson

Jamie Henderson
Food Program Manger

para los niños
for the children

845 East 6th Street
Los Angeles, CA 90021
(213) 623-8446

EXECUTIVE Pa

EXECUTIVE DIRECTOR
Gladys Cabrera

PRESIDENT
Tanya Tull

CHAIRMAN OF THE BOARD
Walter D. Eichner

BOARD OF DIRECTORS
David Blanco
Stephen M. Brown
Leanne Collett
Martha de la Torre
Rudy De Leon
Marjorie S. Green
Wanda Henderson
John Lopez
Antonio Madi
David Morse
Jane Sherwood
Allen N. Smith
Helen Stathatos

ADVISORY BOARD
Tom Bradley
Mayor

Richard Alatorre
Councilman

Jerry Cremins
President, State Building & Construction Trades Council

Stephen Gavin
Former Chairman
Central City Association

John Greenwood
L.A. School Board

Kenneth Hahn
Supervisor, L.A. County

Gilbert Lindsay
Councilman

Robert E. Marciel
Bus. Treas.
Teamsters Local 63

Richard Polanco
Assemblyman

William R. Richardson
Exec. Sec. Treas. L.A. County
Fed. of Labor - AFL-CIO

Edward Roybal
Congressman

Art Torres
State Senator

Cornelius Wall
Reg. Director - ILGWU

Diane E. Watson
State Senator

COUNSEL
Gilin, Scott, Alperstein & Glantz

February 9, 1987

Teresa Masters
President
LOS ANGELES 99'ER COMPUTER GROUP
P.O. Box 3547
Gardena, CA 90247

Dear Ms. Masters:

On behalf of the children, families and staff of PARA LOS NINOS, I wish to thank you and LOS ANGELES 99'ER COMPUTER GROUP for your $25.00 donation.

Your generosity has made life a little brighter for the children of Skid Row and the Central City of Los Angeles.

Your contribution comes at a very critical time for us. We are very much in need. Our survival at this time is dependent on outside support - especially in the private sector. Thank you for being a a part of that support.

Warmest regards,

Erica Reiter

Erica Reiter
Director of Development

ER/slh

THANK YOU FOR CARING!